Making the Case: A New Approach for GIS-Based Geocoding?

Contributed by Bert Granberg
24, Nov. 2010
Last Updated 02, Dec. 2010

This post attempts to describe the current implementation and proposed future implementation of Utah's statewide geocoding web service available via mapserv.utah.gov.

As this is for discussion purposes, please provide feedback via commenting function at the bottom of this post or email to bgranberg@utah.gov.Utah's Statewide Geocoding Web Service: Current Situation

Currently, AGRC provides a statewide geocoding web service that is able to find:
 - Street Addresses, if either a zipcode or a city name is supplied. In this case, the web service input parameters are

 - a registered user name
 - the street addres in standardized format
 - an address zone, either a zip code or a city name** if a city name is supplied, a lookup table is consulted so the application knows the zipcodes in which to look for the address...performance is slower for this option
 - Milepost Locations, if a UDOT-formatted route is supplied. In this case, the same web service is suppled the following input parameters:
 - a registered user name
 - a valid milepost location, with up to 3 significant digits (ex. 123.333 is equal to 1/3 of the way between mileposst 123 and 124)
 - a UDOT-formatted route numbe, the route number with a 'P' at the end (ex. I-15 would be 15P and US89A would be 89AP)

While this web service is utilized by a number of state govenment applications, it has several shortcomings. These are:
 - The service is currently only able to find matches against the statewide road centerline map layer that carries address range information and zipcodes.
 - The service relies heavily on the user supplying the correct zip code or city name which can be confusing for a number of reasons including: imperfect knowledge of zip codes, changing boundaries, postal service prefered placename not agreeing with local place vernacular, special postal service delivery point zip codes for large organizations, etc.
 - When a match is returned, it is impossible to communicate to the end user whether the address is 'hypothetical' or whether, in fact, the address has been officialy assigned to a physical structure or location. In other words, if 120 S Main is found, is there a building or a lot that actually carries that address making it a valid location for a registering voter or driver, or is it just something that someone conjured up out of thin air.Utah's Statewide Geocoding Web Service: Improvement Plan

Expand map layer reference data resources that are used to find an address to include all of the following reference data layers:
 - Address points* (where available)

 - Road centerlines* with address coordinate system attributes to be used for matching zone
 - Parcel polygons* with address attributes**
 - USPS delivery point or post office box zipcode point locations
 - PolylineM linear referencing system for UDOT routes * two levels of address aliasing are available to handle alternative street names** parcels may be second phase or may not be needed if all are represented in address points

Utilize address coordinate system attributes for zones on address points, parcels, and road centerlines. Use a lookup table to translate the user supplied input zone (either a placename ex: 'Murray' or a zip code ex: '84047') into the addressing zone that is utilized in the users area of interest (in this case, both input example 'map' to the address zone 'Salt Lake City' as they use the address origin and grid defined for Salt Lake City. The benefit of adding this extra step to the back end logic is the ability to find the correct location even when the incorrect placename or zipcode is supplied by the user. As long as the user supplies any zipcode or placename associated with the Salt Lake City address zone, the service will find and return the location.

[Draft] Decision tree logic with overload option. When input parameters are supplied to the geocoding service, the service will use map layer resources in a defined default order to find the match unless the calling application provides an overload parameter that specifies a specific resource to use. The web service first must determine if the input parameters are looking for a route and milepost match. If yes, than only data source #5 (above is used). If not, then data sources 1 - 4 are used the order presented above until a satisfactory match is made (score = 100) and the best candidate is returned as the geocoding result.

Numeric Address Transposition (local issue only): Incorporate basic logic to check for reversed numeric addresses. In Utah and limited areas of surrounding states), numeric addresses often specify an address location against an address grid. For example, a typical address might be 1544 S 650 E where 1544 is the house number and the street name is 650 and the street suffix direction is E or East. As these type of addresses specify a location on the grid, the house number component and streetname component are occasionally transposed (650 E 1544 S) which is understood as the same location by locals but not by the geocoding engine. Simple logic should check for this where possible (streetnames almost always end in a 0 or a 5) and untranspose the address components before submitting them to the geocoding resources. The untransposed address for the matching location would be returned along with the location's coordinates.

Other Thoughts:
 - Should there be a service that uses USPS conventions for prefixes (only where they are necessary) and also a service that always expects a directional prefix?
 - Should there be a way to provide both a placename and a zipcode so that both can be tried if necessary?

Test cases:Test Street Address
Test Zone
Test Category
TestName
 Match Method
910 N 300 E
84501ResourcePointFind Point_SName253 N Redwood Rd84116ResourceRangeFind Range_SNamePO Box 322
84114ResourceDeliveryPointFind DeliveryPt40.4556PResourceRouteMilepostSimple1DigitFindUDOT_LRS355.22
15PResourceRouteMilepostSimple2DigitFindUDOT_LRS5.22
313PResourceRouteMilepostSimple3DigitFindUDOT_LRS1.22
89APResourceRouteMilepostException_LRSFindUDOT_LRS251 N 1700 W 84116ResourceSub
ACSAliasFind
Range_SName251 N Highway 68 84116ResourceSub
Alias1Find
Range_SName??? ???ResourceSub
Alias2Find
Range_SName253 N Redwood Rd
Salt Lake CityZoneLookupCityToAddressGridAliasRange_SName253 N Redwood RdMurrayZoneLookup
WrongCityMapsToCorrectAddressGrid Range_SName253 N Redwood Rd
84105
ZoneLookup
WrongZipMapsToCorrectAddressGrid
Range_SName

Geocoder_GeocodesWithPointSName_whenAddressIs_310N300E